

REMARKS

Claims remaining in the present application are Claims 1-16 and 27-37.

CLAIM REJECTIONS

35 U.S.C. 102(e)

The rejection states that Claims 1, 3-9, 27, 28, 30-32, 34, and 37 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,817,005 to Mason et al. (hereinafter, "Mason"). Applicants assume herein that Claim 36 was inadvertently left out of this portion of the rejection. The rejection is respectfully traversed, for the reasons below. It is respectfully submitted that Claims 1, 3-9, 27-28, 30-32, 34, and 37 are not anticipated or rendered obvious by Mason, for the reasons below.

Claim 1 recites:

A method of facilitating a circuit design to be implemented in a programmable device, said method comprising:

a) causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display performed in response to said module being selected;

b) a computer program determining a valid position for said module in a graphical user interface, said graphical user interface having a plurality of resource icons representing said programmable resources, said valid position based on characteristics of said module and characteristics of said programmable resources, said determination made in response to a user request for said valid position for said module in said graphical user interface; and

c) providing in said resource icons an indication of said valid position of said module, said computer program generating said indication.

Claim 1 recites, in part, "a computer program determining a valid position for said module in a graphical user interface, ... said determination made in response to a user request for said valid position for said module in said graphical user interface." Claim 1 also recites "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program."

Mason, as understood by Applicants, may suggest a program for circuit design (Abstract) and may further suggest displaying information related to a module of a plurality of modules where the modules represent a function that is implementable in programmable resources available in the programmable device where the display is performed in response to the module being selected (col. 7, lines 1-40; col. 8, lines 45-55).

However, Mason does not teach the claimed limitations of "a computer program determining a valid position for said module in a graphical user interface" nor does Mason teach the claimed limitations of "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program" as claimed.

The rejection states that Mason cites element "b)" of Claim 1 in column 5, lines 45-53. Mason, as understood by Applicant, may suggest (in col. 5, lines 45-53)

constraints including specifications regarding routing, timing, area mapping and placement, a Constraints editor to enter design constraints, a DLL and an EDIF. However, Applicants are unable to locate that portion within the cited reference that suggests or describes a computer program determining a valid position of a module as claimed above.

Further, Mason, as understood by Applicants, suggests (col. 5, lines 57-62) a Floorplanner tool for floorplanning, and floorplanning which Mason describes as manually placing one or more logic elements and/or blocks of logic into a PLD (Programmable Logic Device). Thus, as understood by Applicant, Mason describes a program for circuit design in which the designer is required to manually place logic blocks. This teaches away from the claimed limitation of Claim 1 which, in part, recites “a computer program determining a valid position for said module in a graphical user interface.”

Continuing, the rejection states that Mason teaches element “b)” of Claim 1 in col. 7, lines 5-57; Figure 1A; col. 8, line 45 to col. 9, line 30. Applicants have read the cited sections and are unable to locate the portion thereof that recites the limitation of element “b)” in Claim 1.

Applicants understand Mason to suggest a program for circuit design in which a floorplanner tool, e.g., a graphical tool, gives the designer control over locating

elements of a logic design in a target PLD using a “drag and drop” paradigm with the mouse pointer (col. 7, lines 5-8). Additionally, Mason, as understood by Applicants, suggests a window in which is displayed the floorplan of the target PLD to allow the designer to draw rectangles into which the designer places logic (col. 7, lines 11-14). Further, Mason suggests the designer (team leader) use the floorplanner tool to manually position each module, the interface circuitry and the global resources/signals within the target PLD. Mason also suggests that final implementation may require additional constraints that may not have been initially envisioned which can affect certain module implementations (col. 7, lines 48-51).

Thus, Mason does not teach the claimed limitation of “a computer program determining a valid position for said module in a graphical user interface” as required in Claim 1. In fact, Mason, as understood by Applicants, describes provisions for correcting design and/or constraint conflicts and/or errors (col. 7, lines 51-57) which teach away from Claim 1.

With reference to Figure 1A and col. 8, line 46 to col. 9, line 30, Applicants are unable to locate that portion within the figure or within the related specification that suggests or describes the claimed limitation of element “b).”

As understood by Applicant, Mason suggests in step 103 (col. 9, lines 18-22) the design is roughly divided into modules where each module includes a high level

description of what each module should do, what region of the target architecture that it should occupy, and necessary connections to other modules. Further, Mason, as understood by Applicants, suggests in step 109 (col. 10, lines 50-65) the floorplanner is used. As described above, the floorplanner needs designer input for determining a module size and its area on the target architecture into which the module is to be implemented. Applicants further understand Mason to suggest a floorplanner that may be able to estimate size and shape of a module or may be able to recommend a size and shape. However, as understood by Applicants, Mason does not teach or suggest the claimed limitation of element "b)" of Claim 1.

Thus, as understood by Applicants, Mason does not teach the claimed limitation of "a computer program determining a valid position for said module in a graphical user interface" as required in Element B.

The rejection further states that Mason teaches element "c)" in Figure 1E and within col. 11, line 66 to col. 12, line 33. Applicants are unable to locate that portion within the cited figure and section in the specification that teaches the claimed limitation of element "c."

With reference to Figure 1E and associated specification and as understood by Applicants, Mason may suggest a graphical interface in which is displayed floorplanned objects connected to respective ports. Thus, by virtue of the objects

being floorplanned and as understood by Applicants, Mason suggests a display of objects in which locations of objects were determined and performed by a designer.

Mason also suggests that the objects, upon selection, are those that are highlighted. However, as understood by Applicants, Mason does not teach highlighting an object to indicate a valid placement of the object, as required in Claim 1.

Further, Mason also suggests “rubber banding” objects to optimize objects relative to related objects within the floorplanned display. Also, Mason may suggest the floorplanner tool generating a top-level file that is used in placement of the object in further processes (Figure 2A).

However, as understood by Applicants, Mason does not teach the limitation of element “c)” of Claim 1. Applicants respectfully assert that further processes in the circuit design as described by Mason are predicated on floorplanning determined and performed by a designer. Accordingly, further processes are subject to error based on erroneous input, as is substantiated in Mason.

Applicants respectfully refer to, including associated specification sections, step 302 of flowchart 300 Figure 3A, step 402 of flowchart 400 of Figure 4, and step 805 of flowchart 800 of Figure 8A, in which Mason, as understood by Applicants,

suggests decisions made regarding corrections and implementations of the object are made. Applicants assert that by virtue of Mason describing an implementing decision in the above mentions steps, with particular reference to step 805 of flowchart 800 (col. 18, lines 15-40), that in a process for final implementation Mason infers that errors can still be present.

Thus, by "a computer program determining a valid position for said module" and "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program," as claimed, embodiments of the instant specification provide an advantage over Mason by substantially reducing if not all but eliminating the instances of design error.

Thus, as understood by Applicant, Mason does not teach or appreciate the claimed limitation of element "b)" of Claim 1.

Therefore, Claim 1 is not anticipated or rendered obvious by Mason. Claims 3-9 and 27 depend from Claim 1, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 3-9 are not anticipated nor rendered obvious by Mason. Allowance of Claims 3-9 and 27 is earnestly solicited.

Further, Claim 28, having similar limitations to Claim 1, and Claims 30-32, 34, and 37 being dependent therefrom, are also allowable for the foregoing

reasons. As such, it is respectfully submitted that Claims 28, 30-32, 34 and 37 are not anticipated nor rendered obvious by Mason. Allowance of Claims 28, 30-32, 34 and 37 is also earnestly solicited.

35 U.S.C. 103(a)

Claims 2, 10-12, 14, and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Mason in view of Publication No. US 2002/0099863 to Comeau et al. (hereinafter, "Comeau"). The rejection is respectfully traversed for the reasons below. It is respectfully submitted that Claims 2, 10-12, 14, and 29 are not rendered obvious by Mason in view of Comeau, for the reasons presented below.

Claim 2

Claim 2, in part, recites:

d) said computer program generating at least two elements selected from the group consisting of: an application programming interface (API) for programming an operation of said module, source code for realizing said module in said resources, an interrupt vector table having a call to an interrupt service routine for said module, and a data sheet for a circuit comprising selected modules as positioned in said graphical user interface.

Claim 2 requires a computer program generating at least two elements from the group consisting of: an application programming interface for programming an operation of a module, source code for realizing the module in the resources, an interrupt vector table having a call to an interrupt service routine for the module, and

a data sheet for a circuit comprising selected modules as positioned in said graphical user interface (emphasis added).

Applicants respectfully assert that Mason and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of “generating an application programming interface (API) for programming an operation of a first of said selected modules.” On page 5, the rejection concedes that Mason fails to disclose this limitation. Moreover, Applicants respectfully submit that Mason fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

Comeau, as understood by Applicants, describes in Paragraph [0046] a peripheral application interface 304 as a structure within support layer 202 to provide application developers a consistent view and communicate with peripherals and processors.

However, Comeau, as understood by Applicant, does not alone or in combination with Mason, suggest generating an application programming interface for programming an operation of a user selected module, as recited in Claim 2.

Moreover, Applicants respectfully assert that Mason and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of “generating an interrupt

vector table having a call to an interrupt service routine for a first of said selected modules.” On page 5, the rejection concedes that Mason fails to disclose this limitation. Moreover, Applicants respectfully submit that Mason fails to teach or suggest this limitation. Comeau also fails to teach or suggest this limitation.

Comeau, as understood by Applicant, describes at Paragraph [0035] a plurality of interrupt vector tables (314, 316) in which processor interrupt vector table 316 points to software support interrupt vector table 314 when a virtual machine 120 and a support layer 202 are compiled together with a bootloader 318. Further understood by Applicants, Comeau, at Paragraph [0036], describes software support interrupt vector table 314 and processor interrupt vector table 316 in which 314 overlaps 316. Comeau, as understood by Applicants and as stated in the Office Action, describes at Paragraph [0049] a scheduler for controlling threads to an interrupt service routine. However, Comeau, as understood by Applicants, does not teach or suggest either generating an interrupt vector table or generating an interrupt vector table having a call to an interrupt service routine, as claimed in Claim 2.

Thus, as understood by Applicants, Mason, in view of Comeau, does not, alone or in combination, suggest generating an interrupt vector table, nor does Mason, in view of Comeau, alone or in combination, suggest generating an interrupt vector table having a call to an interrupt service routine, as claimed in Claim 2.

For the above reasonings, Applicants assert that the teachings of Comeau do not remedy the shortcomings of Mason. Therefore, the combination of Mason in view of Comeau fails to teach or suggest the claimed limitation of Claim 2.

Applicants further traverse the rejection on the grounds that one of ordinary skill in the arts would not have been motivated to combine the teachings of Mason with the teachings of Comeau in such a way as to realize the claimed invention.

Mason describes, as stated by the rejection, a program for circuit design. As understood by Applicants, Mason describes utilizing a high level language, e.g., VHDL (Very ((High Speed Integrated Circuit)) Hardware Description Language) (col. 6, lines 50-53), which inherently implements multiple compilers, e.g., a schematic compiler, a design compiler, a VHDL compiler, and a logic compiler, to enable performing of many of the functionalities contained therein. Thus, as understood by Applicants, Mason suggests an environment for compiled language applications.

Comeau, as understood by Applicants, describes a software support layer for processors executing interpreted languages (Title) with specific mention toward application programmers (Abstract). Comeau does not teach or suggest a software support layer implemented in a circuit designing environment. Further, it is well known in the art that a compiled programming language, such as that suggested by Mason, is well suited for high volume and intense computational functions, e.g.,

circuit design and implementation. It is also well known that an interpreted programming language, e.g., Java, will run slower than a compiled programming language. Thus, Applicants assert that combining the teachings of Mason with the teachings of Comeau would have a detrimental effect to the functions desired by Mason. Thus, Applicants respectfully traverse the cited motivation to combine the teachings of Mason with the teachings of Comeau.

Claim 10

Claim 10 recites:

A computer readable medium having stored thereon program instructions for implementing a method for assisting circuit designing, said method comprising:

a) determining valid positions in a graphical user interface for user selected modules to be placed in said graphical user interface, said graphical user interface describing resources in which said selected modules are programmably operable, said valid positions based on characteristics of said user selected modules and characteristics of said resources; and

b) generating at least two elements selected from the group consisting of: an application programming interface (API) for programming an operation of a first of said user selected modules, source code for realizing said user selected modules in said resources, an interrupt vector table having a call to an interrupt service routine for a first of said user selected modules, and a data sheet for a circuit comprising said user selected modules as positioned in said graphical user interface.

Claim 10 recites, “determining valid positions in a graphical user interface for selected modules to be placed in said graphical user interface, said graphical user interface describing resources operable and in which are implementable said selected modules.” For reasons discussed in the response to Claim 1, Mason fails

to teach or suggest this claimed limitation. Comeau fails to remedy this deficiency. Therefore, neither Mason nor Comeau, alone or in combination, teach or suggest this claimed limitation. As such, Claim 10 is not rendered obvious over the cited combination.

Claim 10 further recites that at least two of the following elements are generated: 1) an application programming interface (API) for programming an operation of a first of said selected modules; 2) source code for realizing said selected modules in said resources; 3) an interrupt vector table having a call to an interrupt service routine for a first of said selected modules; and 4) a data sheet for a circuit comprising said selected modules as positioned in said graphical user interface.

Neither Mason nor Comeau teach or suggest, alone or in combination, generating an API application programming interface (API) for programming an operation of a first of said selected modules, an interrupt vector table having a call to an interrupt service routine for a first of said selected modules, or a data sheet for a circuit comprising said selected modules as positioned in said graphical user interface as claimed. Therefore, Claim 10 is not rendered obvious by Mason in view of Comeau. Continuing with the response to the rejection of Claim 10, the Applicants present the following arguments.

Applicants respectfully assert that Mason and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of “generating an application programming interface (API) for programming an operation of a first of said selected modules.” Applicants respectfully submit that Mason fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

In support of the rejection, Comeau is cited at paragraph [0043]. On page 5, the rejection asserts that Comeau recites an apparatus that supports processors executing interpreted language applications that make use of an API. While Comeau may disclose the use of an API, Applicants assert that Comeau does not teach or suggest the generation of APIs, as claimed.

Thus, Applicants respectfully submit that even if Comeau were to be combined with Mason, the combination fails to teach or suggest “generating an application programming interface (API) for programming an operation of a first of said selected modules,” as claimed.

Moreover, Applicants respectfully assert that Mason and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of “generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules.” The rejection concedes that Mason fails to disclose this

limitation. Moreover, Applicants respectfully submit that Mason fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

In support of the rejection it is asserted that Comeau makes use of interrupt vector tables and routines. However, the rejection fails to assert that Comeau generates such items. Applicants note that Comeau discloses an interrupt vector table. However, Comeau fails to teach or suggest "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules," as claimed. Thus, Applicants respectfully submit that even if Mason were to be combined with Comeau, the combination fails to teach or suggest, "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules," as claimed.

For the foregoing reason, alone or in combination, fail to teach or suggest generating at least two elements selected from the group. Hence, Claim 10 is not rendered obvious by Mason in view of Comeau. As such, Applicants earnestly request allowance of Claim 10.

Claims 11, 12, and 14 depend from Claim 10, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 11, 12, and 14 are not rendered obvious by Mason in view of Comeau. As such, allowance of Claims 11, 12, and 14 is respectfully solicited.

Claim 29

For the reasons discussed in the response to Claim 10, neither Mason nor Comeau, alone or in combination, teach or suggest the claimed limitations of Claim 29. Therefore, allowance of Claim 29 is earnestly solicited.

35 U.S.C. 103(a)

Claims 13, 15, 16, 33, 35, 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Mason in view of Comeau and further in view of Zizzo (U.S. Patent No. 6,578,174). It is respectfully submitted that Claims 13, 15, 16, 33, 35, 36 are not rendered obvious by Mason in view of Comeau further in view of Zizzo, for the reasons presented below.

Applicants respectfully assert that the cited combination does not teach or suggest the claimed limitation of Claim 13 which, in part, claims "d1) determining said new valid position for said first of said user selected modules in said graphical user interface, based on an Extensible Markup Language (XML) description of said first of said user selected modules and an XML description of said resources."

The rejection concedes that Mason in view of Comeau fails to teach the use of XML description of modules, or HTML datasheets of the modules. Applicants

respectfully assert that Mason in view of Comeau does not suggest, describe or teach the claimed limitation.

On page 6, the rejection asserts that Zizzo cites using XML and HTML (column 7, lines 47-52; column 9, lines 20-36). However, the rejection fails to assert that Zizzo determines a new position of a module using XML descriptions of the module and the resource in which the module is to be implemented. Applicants note that Zizzo discloses Extensible Markup Language. However, Zizzo fails to teach or suggest using XML descriptions of modules and resources to determine valid positions of the modules in the resources, as claimed. Thus, Applicants respectfully submit that Mason in view of Comeau in further view of Zizzo does not, alone or in combination, suggest, teach, or describe "determining said new valid position for said first of said user selected modules in said graphical user interface, based on an Extensible Markup Language (XML) description of said first of said user selected modules and an XML description of said resources."

Therefore, Claim 13 is not rendered obvious by Mason in view of Comeau further in view of Zizzo. As such, Applicants respectfully request allowance of Claim 13. Claims 15 and 16 depend from Claim 10, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 15 and 16 are not rendered obvious by Mason in view of Comeau in further view of Zizzo. Allowance of Claims 15 and 16 is respectfully solicited.

Claims 33, 35, and 36 depend from Claim 28, which is believed to be allowable for the aforementioned reasonings with reference to Claim 28. As such, it is respectfully submitted that Claims 33, 35, and 36 are not rendered obvious by Mason in view of Comeau in further view of Zizzo. Allowance of Claims 33, 35, and 36 is respectfully solicited.

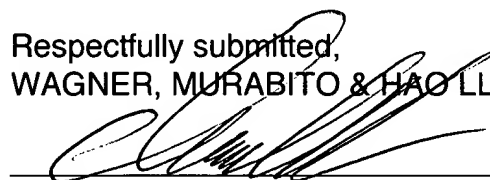
CONCLUSION

In light of the above listed amendments and remarks, reconsideration of the rejected Claims is requested. Based on the arguments and amendments presented above, it is respectfully submitted that Claims 1-16 and 27-37 overcome the rejections of record and, therefore, allowance of Claims 1-16 and 27-37 is earnestly solicited.

Should the Examiner have a question regarding the instant response, the Applicants invite the Examiner to contact the Applicants' undersigned representative at the below listed telephone number.

Dated: 4/22, 2005

Respectfully submitted,
WAGNER, MURABITO & HAO LLP


Anthony C. Murabito
Registration No. 35,295

Address: WAGNER, MURABITO, & HAO LLP
Two North Market Street
Third Floor
San Jose, California 95113
Telephone: (408) 938-9060 Voice
(408) 938-9069 Facsimile